**Introduction**

This report addresses the design challenge of creating a basketball shooting machine that can efficiently return the ball to the free-throw line shooter. The primary focus is determining the optimal combination of launch angle and velocity for easy catching. This endeavor carries significance in basketball training and development, as it can enhance player skills, ensure practice repeatability, save time, cater to various skill levels, and potentially offer performance insights through data collection. Identifying the right launch parameters for the shooting machine is pivotal in advancing basketball training equipment.

**Methods**

The MATLAB program simulates the trajectory of a basketball in the absence of friction. It employs simplified physics assumptions, including constant acceleration due to gravity (g = 9.81 m/s^2) and a fixed time step (dt = 0.01 s). As mentioned earlier, adjusting the initial values of Vx (horizontal velocity) and Vy (vertical velocity) in the simulation code helps illustrate the optimal pass on the y vs. x plot by allowing you to explore a range of possible trajectories for the basketball. In this report, we assume that I am the shooter, in which my shoulders are 150 cm off the ground. We also assume that I am standing 4.572 m away from the initial point where the ball will be shot from. If we want the ball to reach below shoulder length, we simplify this by subtracting 20 cm from the height of my shoulders from the ground to be our target height. This will be 130 cm (150 cm – 20 cm) or 1.3 m.

The program initializes arrays for tracking position (x and y) and velocity (Vx and Vy) and iteratively computes these variables over time using basic kinematic equations, with the use of while loops. The simulation progresses through several tasks, culminating in the determination of launch parameters for the ball to reach the free throw line and be caught at a target height (shoulder level). Refining of the initial conditions by increasing the initial y velocity and adjusting the initial x velocity to replicate a basketball's launch from the same starting point (0.4 m). The simulation proceeds until the ball reaches the free throw line (4.572 m), at which point it calculates the target height for catching (shoulder level - 0.2 m). The program identifies and marks the exact point where the ball reaches this target height on the trajectory plot, or the point closest to it. This approach enables comprehensive analysis of basketball trajectories and facilitates optimization of launch conditions while highlighting the intricate relationships among critical variables during projectile motion.

**Results**

All figures maintain the same initial vertical (y(1)) and horizontal (x(1)) distances in meters (m) but have different configurations of initial vertical and horizontal velocities (m/s), Vy(1) and Vx(1), respectively. The denoted target height in these figures is the point closest to y = 1.3 m.

Figure 1. Vx(1) = 7.5, Vy(1) = 4.5

Figure 2. Vx(1) = 20.5, Vy(1) = 1.5

Figure 3. Vx(1) = 20.5, Vy(1) = 0.5

A graph of a basketball goal

Description automatically generatedA graph of a basketball game

Description automatically generatedA graph of a basketball game

Description automatically generated

Figure 3

Figure 1

Figure 2

**Discussion**

As previously discussed, altering Vx and Vy directly impacts the y vs. x plot by modifying the shape and trajectory of the plotted curve. Adjusting Vx primarily influences the curve's steepness or slope, affecting the horizontal distance covered, while varying Vy influences both the maximum height attained and the overall trajectory's shape. To maintain a constant launch point at a height of 0.4 m, it's imperative that we adjust our Vx and Vy appropriately. This adjustment should strike a balance between feasibility (avoiding excessive speed) and ensuring that no additional movement is required to catch the projectile, given that we are initially 4.572 m away from the launch point. Examining Figure 1, it depicts an optimal trajectory where Vx(1) = 7.5 m/s and Vy(1) = 4.5 m/s. These velocities are neither excessively fast nor too slow, allowing the catcher to remain stationary as the ball is caught approximately 4.5 m away from the shooter, at a height of 1.3 m. This closely matches the constraints of the system. In contrast, Figure 2 exhibits a more arched trajectory, however, we must consider the y-axis range. The maximum height achieved in this case is about 0.5 m, which falls 0.8 m short of our target height. Additionally, at 4.572 m from the starting point, the ball is only 0.48 m above the ground, instead of the desired 1.3 m. This trajectory considers values of Vx(1) = 20.5 m/s and Vy(1) = 1.5 m/s, where the vertical velocity is too low to reach the optimal height, and the horizontal velocity is excessively high. Moving on to Figure 3, this trajectory is suboptimal, as it features a significantly lower vertical velocity of 0.5 m/s. As evident in the plot, there is minimal upward vertical movement, making it challenging to reach our desired height. The analyses conducted in Figures 2 and 3 assisted in determining the optimal velocities displayed in Figure 1. Furthermore, if we aim to calculate the optimal launch angle for achieving these initial conditions, we can utilize the arctan(Vy(1)/Vx(1)) formula, as implemented in my program. The results obtained from the simulation provide a clear understanding of how variations in the initial velocities (Vx and Vy) influence the trajectory of the projectile and its ability to meet specific constraints. To enhance the accuracy of the simulation, several improvements can be considered. Firstly, fine-tuning parameters like air resistance, gravity, air density, wind speed, surface friction, and launch angles to closely match real-world conditions can enhance precision. Additionally, collecting a greater number of data points during the simulation can provide a more detailed representation of the projectile's path. Collecting data from a controlled system can also help improve the discrepancies between experimental and predicted/simulated results.